NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A GEOGRAPHIC OPTIMIZATION APPROACH TO COAST GUARD SHIP BASING

by

Mehmet Ali Gençay

June 2015

Thesis Advisor: Roberto Szechtman
Second Reader: Michael Atkinson

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 2015	Master's Thesis 06-23-2013 to 06-19-2015

4. TITLE AND SUBTITLE

A GEOGRAPHIC OPTIMIZATION APPROACH TO COAST GUARD SHIP BASING

5. FUNDING NUMBERS

6. AUTHOR(S)

Mehmet Ali Gençay

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense, the U.S. Government, or the Turkish Coast Guard. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

This thesis studies the problem of finding efficient ship base locations, area of operations (AO) among bases, and ship assignments for a coast guard (CG) organization. This problem is faced by many CGs around the world and is motivated by the need to optimize operational outcomes in the face of budget constraints. There is a need for a tool to optimize the placement of the available number of ships to candidate bases and the assignment of the AOs for each base. In this thesis, we developed a model that takes the objective of minimizing the weighted demand for CG services. We also used constraints to have proportionate AOs to the number of ships on each base and partition constraints. To the best of our knowledge, until now, there has yet to be a tool designed for finding both ship allocations and AOs for each base. Therefore, developing this tool is a huge step forward in this area.

14. SUBJECT TERMS

coast guard, ship basing, ship allocation, geographic optimization

15. NUMBER OF PAGES 69

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

A GEOGRAPHIC OPTIMIZATION APPROACH TO COAST GUARD SHIP BASING

Mehmet Ali Gençay
Lieutenant Junior Grade, Turkish Coast Guard
B.S., Turkish Naval Academy, 2009

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 2015

Author: Mehmet Ali Gençay

Approved by: Roberto Szechtman
Thesis Advisor

Michael Atkinson
Second Reader

Robert F. Dell
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis studies the problem of finding efficient ship base locations, area of operations (AO) among bases, and ship assignments for a coast guard (CG) organization. This problem is faced by many CGs around the world and is motivated by the need to optimize operational outcomes in the face of budget constraints. There is a need for a tool to optimize the placement of the available number of ships to candidate bases and the assignment of the AOs for each base. In this thesis, we developed a model that takes the objective of minimizing the weighted demand for CG services. We also used constraints to have proportionate AOs to the number of ships on each base and partition constraints. To the best of our knowledge, until now, there has yet to be a tool designed for finding both ship allocations and AOs for each base. Therefore, developing this tool is a huge step forward in this area.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

List of Acronyms and Abbreviations

AO area of operation

AoA analysis of alternatives

CG Coast Guard

CGSB coast guard ship basing

nm nautical miles

NPS Naval Postgraduate School

TCG Turkish Coast Guard

USCG United States Coast Guard

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

This thesis studies the problem of finding efficient ship base locations, area of operations (AO) among bases, and ship assignments for a coast guard (CG) organization. This problem is faced by many CGs around the world and is motivated by the need to optimize operational outcomes in the face of budget constraints. There is a need for a tool to optimize the placement of the available number of ships to candidate bases and the assignment of the AOs for each base.

In this thesis, we capture the objective of minimizing the sum of the average demand (importance and quantity) multiplied by the distance of the point (x,y) to the assigned base. Minimizing the distance to the assigned base, which is weighted by the importance and quantity of the demand, is appropriate. Some CGs may think that other objectives would be more appropriate, like maximizing the covered demand or minimizing the response time. Thus, anyone can build a new objective and use the remaining constraints of our model to develop his own model.

Having AOs proportionate to the number of ships on each base was the most critical constraint. We controlled the AO sizes with this constraint and equalized the proportion of the coverages in every AO. This also can be altered if so desired.

Other constraints in our model were partition constraints, which are standard in facility location problems, as mentioned in the literature review.

After developing the model, we used it to find AOs for each base given ship allocation, base locations, and demand functions. Then we used a greedy algorithm, which basically tries to get the most utility out of the available options, to find the best ship allocation given base locations and total available number of ships.

To the best of our knowledge, until now, there has yet to be a tool designed for finding both ship allocations and AOs for each base. Therefore, developing this tool is huge step forward in this area.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

I would like to express my appreciation to Associate Professor Roberto Szechtman for the great support and guidance he has provided me. I could not have completed this thesis without his efforts.

I am grateful for the Turkish Coast Guard and the Turkish nation for giving me this huge opportunity to study at NPS and to have wonderful friends from different nations.

A special thanks to my wife for her great sacrifice and patience. She has been always there for me when I needed her the most. Thank you for taking care of me.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

This thesis studies the problem of finding efficient ship base locations, area of operation (AO) among bases, and ship assignments for a Coast Guard (CG). This problem is faced by many CGs around the world and is motivated by the need to optimize operational outcomes in the face of budget constraints. Most CGs have a common set of missions; in this chapter, we provide background on the scope and mission of the CG and the objective of this study.

1.1 Background

In the United States, the goals of the CG include providing free and safe passage for sea traffic, providing the needs of navigation and land transportation, and regulating bridge lighting for the safety of navigation and land traffic [1].

More precisely, the United States Coast Guard (USCG) has 11 missions [2]:

1. Ports, waterways, and coastal security
2. Drug interdiction
3. Aids to navigation
4. Search and rescue
5. Living marine resources
6. Marine safety
7. Defense readiness
8. Migrant interdiction
9. Marine environmental protection
10. Ice operations
11. Other law enforcement

These missions and the task definition of the USCG clearly demonstrate that this organization has many responsibilities and a large operation area, as is the case with almost all of the CG services around the world. Except for the ice operations, all the missions above are standard. These missions are not only important to the government in terms of economic

and political reasons, they are also important in saving the lives of individuals.

Having a large operation area and operating actively on an everyday basis make professional life difficult for the men and women of CGs all over the world. A good example is the CG base in Scituate, Massachusetts, that closes during the winter season due to bad weather; crews are relocated to CG Station Point Allerton in Hull, Massachusetts [3]. It is a seasonal closure of the base but USCG may need an optimization of the available resources for the winter season since the Scituate base is closed during winter. For example, the USCG policy is to return the crew to their home duty station, but the distance from Hull to Scituate is over 10 nautical miles (nm). The second closest base is in Provincetown, Massachusetts, which is over 25 nm. Therefore, if a nearby civilian needed help while the Scituate base was closed, he or she might need to wait additional time. The difference in distance could be a problem in a time-critical situation. Besides that, many waterways are farther than that from the nearest CG station. USCG needs an optimization tool that takes this distance factor into account.

Seamen of the CGs are faced with extremely busy schedules. Despite the danger at sea, people in regions with high levels of conflict seek a better place to live, and immigration by sea is often their favourite choice. As a result, many immigrants have died pursuing their dreams of finding a safe country for their children. In April 2015, 700 migrants almost drowned in the Mediterranean just outside Libyan waters. This incident could have been the worst disaster yet involving immigrants being smuggled into Europe [4]. In another incident the Italian CG rescued nearly 4000 immigrants in just one day from smugglers' boats on the Mediterranean Sea on May 3, 2015 [5].

Additionally the Turkish Coast Guard (TCG) saved 895 immigrants between April 13-19, 2015, according to the TCG's official website [6]. As of April 2015, the TCG had already saved 923 immigrants in five events in 2015 [7]. According to another source, the TCG saved 636 immigrants in just five days in the Aegean Sea [8]. On April 15, 2015, TCG saved 42 people in a single event right outside of Dikili, Izmir, as shown in Figure 1.1 [9].

1.2 Objective

Because of these difficulties and limitations, CG services need to optimize the utility which they are getting out of their resources. We think that there is a need for a tool to optimize

Figure 1.1: Turkish Coast Guard saving immigrants on a CG mission

the placement of the available number of ships to candidate bases and the assignment of the AOs for each base. As of April 2015, the USCG is in the process of developing a homeporting manual to support these types of decisions, according to a USCG program analyst [10].

We want to develop a tool to help decision makers to place ships along their coastlines. Since we need to know the effects of having multiple ship allocations, base locations, demand functions, ship types, and AOs, this problem is complicated to solve. Instead of solving a particular problem, we decided to develop a model to help every decision making process related to coast guard ship basing (CGSB).

1.3 Research Questions

We mainly focused on answering the following questions. First, we try to answer the question of how can we characterize overall performance in terms of base location, ship assignments to each base, and AOs for each base. The second question is what are the optimal base locations, ship allocations, and AO for each base. Finally, how do the answers to the first two questions change when the mission demands are random. We will address these questions in Chapters 4 and 5.

1.4 Scope and Methodology

This research focuses on maximizing an overall expected utility that depends mainly on the response times, the number of unfulfilled missions, and the construction and operational cost of each CG port.

Due to time constraints, we could only study a part of the problem, but it's always possible to have additional constraints to allow our model to capture more. In this study, we dealt with ship allocations and the AOs for each base given the total number of ships and the base locations. First, we developed a model with an objective of minimizing the weighted distance and a constraint of having proportional AOs with the number of ships in each base. The model gives the output of AOs for given base locations and ship allocations. Then, we used a *greedy algorithm*, which tries to get the most utility out of the available options, to find the best ship allocation for given base locations and total available number of ships.

First of all, we need to define the right objective that we want to capture for the CGSB problem. Then, we decide what constraints play a role in the real world and how we should include them to our model. To keep our model understandable, we keep it simple at first. The only outcome of our model is AOs for each base. Then, we can apply the *greedy algorithm* technique to find the optimum ship allocation.

1.5 Thesis Organization

We provide a literature review in Chapter 2, detailed explanation of the model in Chapter 3, analysis of the model in Chapter 4, and the conclusions and the recommendations in Chapter 5.

CHAPTER 2:
Literature Review

Existing literature does not address the operational problem of determining the appropriate location for CG bases. This problem is unique in several ways. First, determining the appropriate location involves the joint objective of placing bases and placing the AO for each base. Second, this placement occurs in a one-dimensional environment along the coastline, not in a two-dimensional environment as is more traditional in the literature. Third, in the industrial world, demand only occurs after placing the facilities. However, demand for the CG is not directly related to the location of the bases. Instead, this demand is based on population, sea traffic, and the nature of the coastline. Fourth, location is constrained by the fact that the entire coastline is not available for basing CG ships. Finally, the assignment of ships to bases needs to be predetermined because the capacity of the facilities is not identical.

In the industrial sector, the facility location problem deals with efficiently placing several facilities in a given area and dividing that area for several vehicles to meet the demands of customers. For this thesis, opening the appropriate bases for CG ships is equivalent to placing several facilities in the industrial sector. Additionally, finding an allocation for the ships is equivalent to dividing the area for several vehicles. In our case, since we can only place bases on the coastline, it is a one-dimensional problem rather than a two-dimensional problem. To the best of our knowledge, there is no work in the existing literature that fully captures the operational situation that is the subject of this thesis.

Since we could not find any information on our particular problem, this literature review focuses on the most relevant problem, which is facility location. We first discuss vehicle routing and then focus on facility location because in the OR community, these two problems are mostly addressed together and can be referred to as inseparable. Vehicle routing is a second-stage problem in facility location, so readers need to be informed about this concept.

2.1 The Vehicle Routing Problem

The vehicle routing problem is a problem where we want to meet the demand from our clients by serving them sequentially while using a route which is cost efficient. This can be an example of how delivery systems work. Mailing companies, postal services, and distribution companies of any sort use these techniques which are referred to as the vehicle routing problem.

A basic example for the vehicle routing is Federgruen and Zipkin [11]. Federgruen and Zipkin consider only the vehicle routing problem and not the facility location problem. Federgruen and Zipkin address the problem of allocating a limited resource among different locations and plan deliveries for the random demands in the area via delivery vehicles. Federgruen and Zipkin also take holding and shortage costs into account along with transportation costs. Federgruen and Zipkin propose an approach that is computationally fast enough for practical work.

Vehicle routing problems can be more complex in terms of capacity restrictions. Carlsson [12] considers the case of an uncapacitated stochastic vehicle routing problem where the locations of the depots are fixed and the locations of the clients are unknown in his paper. Carlsson aims to balance the workloads of all vehicles while dividing the area into smaller regions. He uses a one-stage optimization technique. Carlsson uses a probability function for the demands and using this information found an optimal result for partitioning. Carlsson applies the travelling salesman problem (tries to find the shortest path to visit a list of cities) for the routing problem. Detailed information about vehicle routing and other studies about this topic can be found in Min et al. [13].

2.2 The Facility Location Problem

The facility location problem has been considered in many different ways over the years and people may define different objectives as useful. Locating a facility does not necessarily need to have the same purpose. One may try to minimize their costs, while others may try to minimize the distance from the customers to their facility. Even placing some security cameras for surveillance of a place may be described as a facility location problem. There are a variety of objectives to choose from like minimizing the cost, minimizing the overall distance, maximizing the minimum workload, etc. We want to explain and give some

examples of them.

2.2.1 Objectives of the Facility Location Problems

As an example of minimizing the distance, the Fermat-Weber problem can be a good example. The Fermat-Weber problem is related to the problem of placing several facilities in given area, where the goal is to minimize the distance between facilities and any point given in the area. Its discrete and continuous versions are analyzed in Hamacher and Drezner [14] and Fekete et al. [15].

Minimum equitable radius and the n-center problems have their own unique objectives. In the equitable radius problem the objective is to place n equal area Voronoi cells while minimizing the maximum distance from any point to the closest cell. A good paper on the subject is Suzuki and Drezner [16]. On the other hand, in the n-center problem, the objective is having the minimum radius for n identical circles to cover the area. A well-written paper on the subject is Suzuki and Drezner [17].

Minimizing the maximum workload is another example of the objectives for a facility location problem. Carlsson and Devulapalli [18] study the problem of dividing a geographic region into subregions so as to minimize the maximum workload of facilities in that region. Our work in this thesis is related to Carlsson and Devulapalli, and so we discuss it in more detail. Carlsson and Devulapalli take a function of distance to capture the cost effect of serving from a facility to its assigned region. When the demand points follow a continuous probability density, Carlsson and Devulapalli use a Multiplicatively Weighted Voronoi Diagram to find the boundaries between the regions. They also prove the boundaries found using Multiplicatively Weighted Voronoi Diagram are the optimal boundaries.

Giving an explanation of the Voronoi Diagram will be helpful at this point. A Voronoi Diagram is a geometrical construct which divides the plane according to the *nearest-neighbor rule*: each point is associated with the region closest to it. Voronoi diagrams are described with metrics which are modified by weights. Using only non-negative weights implies positively weighted Voronoi diagrams.

Carlsson and Devulapalli examine versions with and without equal area constraint which either requires every region to have equal area of service after partitioning or not. Another

problem considered by Carlsson and Devulapalli aims to minimize the overall workload with the equal area constraints.

For the problem of having disconnected subregions for a facility, Carlsson and Devulapalli offer two alternatives. The first alternative is having a problem with the objective function of minimizing the overall workload. That would result with the optimal subregions which are connected. That is defined as observed but never proved. We mention this case since we also use the objective function of minimizing the overall workload and we will discuss the disconnected region problem in Chapter 4.

To capture the benefits of max-min problem along with the minimum total workload problem, Carlsson and Devulapalli offer a weighted problem which is having the weighted combination of the objective functions of two problems.

The second alternative to not having disconnected subregions is having a maximum distance constraint between any point x to its assigned facility.

As a further application of their model, Carlsson and Devulapalli also discuss the case where the facility placement and the subdivision of the territory is variable. The authors state their findings and claim a model but they note that the final configuration is in no way guaranteed to be globally optimal.

After mentioning the variety of the objectives used for facility location problem, we should state that having an objective of minimizing the overall cost or maximizing the overall utility is the most common case and the most appropriate objective to our problem.

Minimize the Total Cost

Having an objective of minimizing the total cost surely limits the scope of the research but there are still many different kind of examples in the literature. One of them is multi-commodity, multi-plant. An important article dealing with facility location that considers the problem of multi-commodity, multi-plant, capacitated facility location is Pirkul and Jayaraman [19]. Pirkul and Jayaraman give an algorithm called PLANWAR (PLANt and WARehouse). PLANWAR is a mixed integer programming model and is based on Lagrangian relaxation of the multi-commodity, multi-plant, capacitated facility location problem. Pirkul and Jayaraman also present a heuristic solution to get a good estimate of how

good PLANWAR performs.

In this paper, Pirkul and Jayaraman consider the problem where customers demand multiple units of different kinds of products and there are warehouses to meet these demands. Warehouses are also fed by the multiple plant locations. The objective of this multi-commodity model is to minimize the sum of the fixed costs of having these warehouses and the sum of the transportation costs of both transferring the goods to warehouse and from warehouse to the customers. PLANWAR gives the optimum location of the warehouses and storage of goods in these warehouses as well as what to plant on each of the planting locations.

An example of using a greedy algorithm to the facility location problem is Guha and Khuller [20]. They study the location of the facilities such as warehouses and industrial plants. It is an *uncapacitated facility location* problem, and the driving factor is transportation cost. Their objective function has both transportation and fixed cost of having a facility open. The authors develop a greedy algorithm to solve this NP-hard problem.They conclude that using a "local-improvement" algorithm, along with the algorithm given by Shmoys et al. [21] yields an approximation factor of 2.408 of the optimal.

Shmoys et al. also study the same *uncapacitated facility location* problem, and their polynomial-time algorithm gives a solution to the cost within a factor of 3.16 of the optimal given the distances between locations are non-negative, symmetric and satisfy the triangle inequality.

So far we talked about meeting the demand with a single facility. One might want to be able to meet the demand with k different facilities. We examined Aardal et al. [22] as an example of the k-level problem uncapacitated facility location. k-level uncapacitated facility location problem where every predetermined demand point needs to be serviced by a sequence of k different facilities is considered and there are positive fixed costs for setting up a facility in the paper. There are no capacity restrictions on the facilities and distances are all non-negative and satisfy triangle inequality. The objective is to meet the demand of each customer with k different facilities while minimizing the total setup and service costs.

Aardal et al. [22] find a randomized algorithm which is using the optimal solution of a linear programming relaxation and its dual in order to make random choices. This algorithm is

proven to find a feasible solution of expected cost within factor of 3 of optimum cost, although they have not performed the algorithm in practice.

As we mentioned before, Shmoys et al. [21] find an algorithm which gives 3.16-approximation and later Chudak [23] advances the algorithm which is capable of giving a solution of 1.736-approximation but all are considered 1-level problem. Whereas Aardal et al. [22] gave the algorithm where $k \geq 2$.

2.2.2 Different Approaches to Facility Location Problem

As an industrial application of the facility location problem we look at Corum [24]. In this paper there are two types of facility planning which are facility location and facility design, and the focus is on the facility location problem. The specific problem analyzed is the location analysis of a bowling alley. The approach is an analysis of alternatives. Three types of factors make up the objective function: the critical factors, the objective factors, and the subjective factors. Critical factors are the most important factors which define whether an alternative should be considered or not. Objective factors are the factors which can be computed quantitatively. Subjective factors are considered between 0 and 1 and a value is given accordingly to evaluate the factors quantitatively. Corum calculates an overall value to compare the alternatives and choose the best option by giving different weights to each factor.

One of the other approaches to the facility location problem can be handling the problems one at a time with a staged model. We mention Haugland et al. [25] as an example. Haugland et al. consider the problem of creating districts for vehicle routing problem with stochastic demands. Haugland et al. use a two-stage optimization technique (partitioning and routing) which is essential because demands are revealed only after the districts are determined. Haugland et al. first partition the area into subregions, and once the demands are revealed they optimize the route for the delivery. Haugland et al. use Tabu search (search technique which always seeks for a new route) and multi-start heuristic (solving with different starting points, but never guaranteed to be the globally optimal) for the problem and his computational results shows that Tabu search is superior to multi-start heuristic. Similarly to this basic idea, we also use a two-staged optimization technique to find optimum locations of the bases for our CG problem.

After a broad introduction to studies which have been done in the facility location area, finally we are going to talk about the main inspiration to our work. Before we do that, we should state Aikens [26] can be a good source for anyone who seeks more information about the problem. Aikens gives a broad perspective on the whole facility location problem in distribution planning. Aikens considers the distribution planning facility location problem, in particular problem of simple incapacitated facility location, the simple incapacitated multi-echelon facility location, the multi-commodity incapacitated facility location, the dynamic incapacitated facility location, capacitated facility location, generalized capacitated facility location, stochastic capacitated facility location, and multi-commodity capacitated single-echelon facility location.

2.3 Main Reference

In the main paper which inspires our work, Carlsson and Devulapalli [27] consider a geographical optimization problem defined in area R which has an objective of balancing the utility over all of the n regions defined in R. These utilities are given by the integrals $\iint_{R_i} f(x)u_i(x)\,dA$. Carlsson and Devulapalli use a probability density function of $f(.)$ and n utility density functions $u_i(.)$ which are defined in R.

Carlsson and Devulapalli define their problem as following to maximize the overall utility:

$$\begin{aligned} \max_{R_1,\ldots,R_n} \sum_{i}^{n} \iint_{R_i} & f(x)u_i(x)\,dA \\ \textbf{s.t.} \iint_{R_i} & f(x)\,dA = q_i \quad \forall i \\ & R_i \cap R_j = \emptyset \quad \forall i \neq j \\ & \bigcup_i R_i = R \end{aligned} \tag{2.1}$$

Here is another optimization problem defined by Carlsson and Devulapalli to have a different approach to the same problem. Carlsson and Devulapalli give this optimization problem

to find a way to maximize the minimum utility of all n subregions.

$$\begin{aligned} \max_{R_1,\dots,R_n} \min_i \Big(\iint_{R_i} f(x)u_i(x)\,dA \Big) & \\ \textbf{s.t. } R_i \cap R_j = \emptyset \quad \forall i \neq j & \\ \bigcup_i R_i = R & \end{aligned} \tag{2.2}$$

Carlsson and Devulapalli take both problems into consideration and solve them separately to give the reader an option to choose the optimization problem which defines their problem. Since we can have different number of ships assigned to each base, we are not concerned about equilizing the utilities among the bases. We find the optimization problem 2.1 is more related to our work and we also used the objective of maximizing the overall utility/minimizing the overall distance for our model.

After taking the dual of the optimization problem 2.1 and with some adjustments, Carlsson and Devulapalli use the following dual problem in their algorithm to solve it.

$$\begin{aligned} \min_\lambda \iint_R f(x) \max_i \{u_i(x) - \lambda\}\,dA & \\ \textbf{s.t. } q^T \lambda = 0 & \end{aligned} \tag{2.3}$$

Carlsson and Devulapalli state that by using complimentary slackness technique, boundary curves that shapes the areas are either functions of $u_i(x) - u_j(x)$ or the ratio of $u_i(x)/u_j(x)$.

We also used the dual of our optimization problem, and find an algorithm to find the solution in Chapter 3.

Carlsson and Devulapalli maximize the utility, while we minimize the distance from the base. They prove to have equal values for functions of $u_i(x) - u_j(x)$ or the ratio of $u_i(x)/u_j(x)$; similarly we have equal values of $u(y) - \lambda_i(u(y)/d_i(y))$ functions for different bases on boundaries. They also use partition constraints that we need in our problem. Having these similarities between our problem and Carsson and Devulapalli and the ease of application of the dual problem, we think that our problem is much more related to their paper than any other paper we mentioned on this subject. That is why we use this paper as

a foundation to our work and we will discuss our model in the following chapter.

After developing a model with these similarities with Carlsson and Devulapalli, we implemented a *greedy algorithm* to find the optimal ship allocation as Guha and Khuller [20] did.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:
Model Development

In this chapter we present the main model, which extends Carlsson and Davulapalli's study [27]. In Section 3.1, we introduce our basic model and explain how we developed it; in Section 3.2, we explain the dual process of our model; in Section 3.3, we explain the way our algorithm works; and in Section 3.4, we give the algorithm used to implement our model.

3.1 Basic Model Formulation

Our goal is to suitably capture the objective of the CG, while at the same time keep the model tractable. In order to do that, we thought about real world problems that the CG encounters. Thusly motivated, we assume that the shore is vertical and follows the y axis, so that the CG bases are located along that axis. Also, we average the demand along the x axis, meaning that we assume that the demand is concentrated at a fixed distance x from the shore and that the value of x is constant in y (see Figure 3.1)

Regarding the demand, we consider two main aspects: the quantity and the importance. The average quantity at the y latitude is described by a real function $q(y)$, and the average importance (or *weight*) at latitude y is represented by the real function $w(y)$. Hence, the overall average demand per day at latitude y is $d(y) = q(y) \times w(y)$. This setup allows us to capture calls of varying importance from vessels like sailboats to the relatively fewer but critically important oil tankers.

The CG selects bases, allocates ships, and divides the AOs between each base so as to minimize the distance to the demand. This has the concrete benefit of minimizing average response times. More specifically, for ships based at fixed location y_i where $i \in \{1, \ldots, n\}$, the goal is to minimize $\int_{\Delta_i} d(y)(x^2 + (y - y_i)^2)^{1/2} dy$, where Δ_i is the width of the AOs of base i (see Figure 3.2) The decision variables that capture the AOs for each base are defined as $\Delta_1, \Delta_2, \ldots, \Delta_n$.

The model we use can be formulated as follows:

$$\min_{\Delta_1,\Delta_2,\ldots,\Delta_n} \sum_i \int_{\Delta_i} d(y)(x^2+(y-y_i)^2)^{1/2}\,dy,$$

$$\int_{\Delta_i} q(y)(x^2+(y-y_i)^2)^{1/2}\,dy \le s_i \quad \forall i \in \{1,\ldots,n\}$$

$$\Delta y_i \cap \Delta y_j = \emptyset \quad \forall i \neq j, \text{ and } i,j \in \{1,\ldots,n,\}$$

$$\Delta y_1 \cup \Delta y_2 \cup \ldots.. \cup \Delta y_n = [0,Y],$$

Considering all bases jointly, the overall objective is to minimize the weighted distance as shown below:

$$\min_{\Delta_1,\Delta_2,\ldots,\Delta_n} \sum_i \int_{\Delta_i} d(y)(x^2+(y-y_i)^2)^{1/2}\,dy,$$

where x is the average location of the demand, for simplicity assumed to be constant in y.

Next, we consider the constraints. We have two types of constraints: those that correspond to the ships' capabilities and those that partition the AO. Regarding the former, for ships assigned to a base located at y_i, the average distance they can cover is limited by the following constraint:

$$\int_{\Delta_i} q(y)(x^2+(y-y_i)^2)^{1/2}\,dy \le s_i \quad \forall i \in \{1,\ldots,n\}$$

for $s_i = n_i \times 200 \times (1/k)$, where s_i is the number of ships in base i (n_i) times 200, which is the total distance that can be made by a CG ship in a day. We chose to use 200 nm because we assumed an average speed of 20 knots and a 10-hour work in a day is appropriate. Since the demand functions are also determined as the average demand in a day, our constraint is consistent. Also, the constant k is a multiplier of the s_i which is the percentage of the demand covered in a given area. Since we are trying to find a solution for different total numbers of ships for the same demand function, we need a parameter (k) to tell us the coverage of the ships.

The parameter k is an input to our model. It is constant throughout the area. It represents the fraction of demand covered by the ships assigned to each base. However we do not know what the true demand met will be ahead of time. If k is large, then the constraint

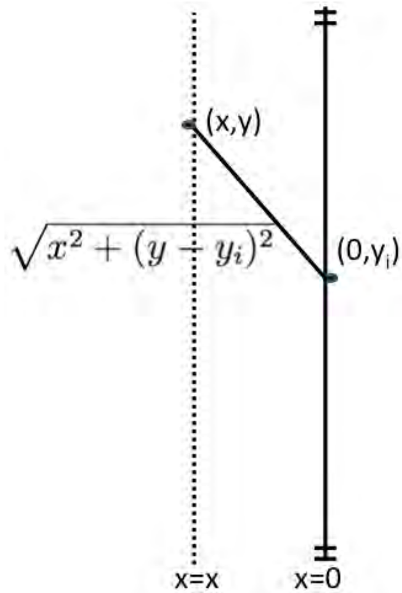

Figure 3.1: Visual representation of the distance function from point (x,y) to base located on $(0,y_i)$ along coastline on y-axis

will be infeasible. Thus, in practice, we start with a small k (ensuring feasibility), and increase it until right before it becomes infeasible. The value of k generated by this process is the coverage achieved by our allocation assignment. We use k in this fashion to examine having different numbers of ships which are not enough to cover the whole AO.

Observe that the *importance* at latitude y, $w(y)$ does not play a role in this constraint because the distance traveled by a ship depends only on the demand quantity times the distance to it.

The partition constraints are standard and given as follows:

$$\Delta y_i \cap \Delta y_j = \emptyset \quad \forall i \neq j, \text{ and } i, j \in \{1, \ldots, n,\}$$

and

$$\Delta y_1 \cup \Delta y_2 \cup \ldots.. \cup \Delta y_n = [0, Y],$$

where the coastline lies along the segment $[0,Y]$. These constraints ensure that there is no overlap in the areas of operation assigned to each base. The entire coastline is covered by ships assigned to exactly one base while in reality, there may exist some overlap in the AOs

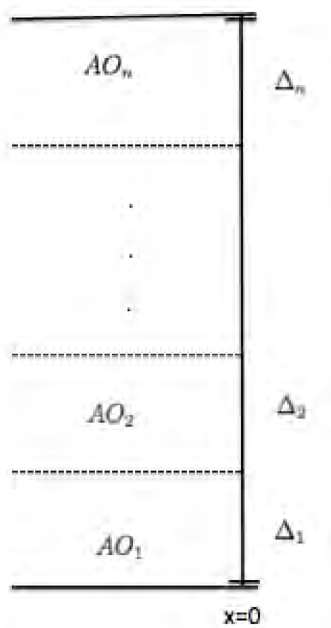

Figure 3.2: Visual representation of the AO, which are limited by the Exclusive economic zone on x-axis, for the bases located along coastline on y-axis

(e.g., for safety reasons). Our formulation can be easily adapted by suitably redefining the partition set.

So far, we assume that the ships are assigned to each base, but this will be relaxed later in this chapter.

3.2 Dual Formulation

In this section we derive the dual problem, which leads to a numerical algorithm.

To wit, set $I_i(y) \in \{0,1\}$ that indicates whether point y is assigned to base i or not. Then, the *primal problem* becomes

$$\min_{I_1,I_2,\ldots,I_n} \sum_i \int_{\Delta_i} d(y)(x^2+(y-y_i)^2)^{1/2} I_i(y)\, dy$$

subject to

$$\int_{\Delta_i} q(y)(x^2+(y-y_i)^2)^{1/2} I_i(y)\, dy \le s_i \quad \forall i$$

$$\sum_i I_i(y) = 1 \quad \forall y \in [0,Y]$$

$$I_i(y) \in \{0,1\} \quad \forall i \in \{1,\ldots,n\}, \text{ and } y \in [0,Y]$$

The linear relaxation of the above problem is

$$\min_{I_1,I_2,\ldots,I_n} \sum_i \int_{\Delta_i} d(y)u_i(y)I_i(y)\,dy$$

s.t.

$$\int_{\Delta_i} q(y)u_i(y)I_i(y)\,dy \leq s_i \quad \forall i \in \{1,\ldots,n\}$$

$$\sum_i I_i(y) = 1 \quad \forall y \in [0,Y]$$

$$I_i(y) \geq 0 \quad \forall i \in \{1,\ldots,n\}, \text{ and } y \in [0,Y],$$

where we simplified the notation by defining $u_i(y) = (x^2 + (y - y_i)^2)^{1/2}$.

We discretize the AOs into cells $j = 1,\ldots,N$ of area ε for average x and appropriate y values, where $z_{i,j}$ is the proportion of mission demands in cell j assigned to base i. Also, let u_{ij}, d_j, and q_j be the average value of their respective continuous variables over cell j. Then, the preceding integer relaxation becomes

$$\min_z \sum_i \sum_j \varepsilon d_j u_{ij} z_{ij}$$

s.t.

$$\sum_j \varepsilon q_j u_{ij} z_{ij} \leq s_i \quad \forall i \in \{1,\ldots,n\}$$

$$\sum_i z_{ij} = 1 \quad \forall j \in \{1,\ldots,N\}$$

$$z_{ij} \geq 0 \quad \forall i \in \{1,\ldots,n\}, \text{ and } j \in \{1,\ldots,N\}$$

The dual of this problem becomes

$$\max_{\lambda,\varsigma} \sum_i s_i \lambda_i + \sum_j \varsigma_j$$

s.t.

$$\varepsilon q_j u_{ij}\lambda_i + \varsigma_j \leq \varepsilon d_j u_{ij} \quad \forall i \in \{1,\ldots,n\}, \text{ and } j \in \{1,\ldots,N\}$$

$$\lambda_i \leq 0 \quad \forall i \in \{1,\ldots,n\}$$

Introducing a new variable $\sigma_j = \varsigma_j/(\varepsilon d_j)$, we obtain

$$\max_{\lambda,\sigma} \sum_i s_i\lambda_i + \sum_j \varepsilon d_j \sigma_j$$

s.t.

$$\sigma_j \leq u_{ij} - \lambda_i \frac{q_j u_{ij}}{d_j} \quad \forall i \in \{1,\ldots,n\}, \text{ and } j \in \{1,\ldots,N\}$$

$$\lambda_i \leq 0 \quad \forall i \in \{1,\ldots,n\}$$

After substituting in $\frac{q_j u_{ij}}{d_j} = \frac{u_i(y)}{w(y)}$, and passing to the limit at $N \to \infty$ and $\varepsilon \to 0$, the *continuous* version of the dual problem becomes

$$\max_{\lambda,\sigma} \sum_i s_i\lambda_i + \int_0^Y d(y)\sigma(y)\,dy$$

s.t.

$$\sigma(y) \leq u_i(y) - \lambda_i \frac{u_i(y)}{w(y)} \quad \forall i \in \{1,\ldots,n\}, \text{ and } y \in [0,Y]$$

$$\lambda_i \leq 0 \quad \forall i \in \{1,\ldots,n\}$$

Finally, the latter is equivalent to the problem

$$\max_{\lambda} \sum_i s_i\lambda_i + \int_0^Y d(y) \times \min_i \left(u_i(y) - \lambda_i \frac{u_i(y)}{w(y)}\right) dy$$

s.t.

$$\lambda_i \leq 0 \quad \forall i \in \{1,\ldots,n\}$$

The dual variables $\lambda_1,\ldots,\lambda_n$ induce a partition by assigning point y to the base that solves $\min_i\{u_i(y) - \lambda_i u_i(y)/w(y)\}$.

To be able to compute the objective, we need to find the $u_i(y) - \lambda_i u_i(y)/w(y)$ lines for each y and assign them to base i where i makes this value minimum. This would be the AO

assignment for the bases. By changing the λs, we shift the lines up and down which allows us to meet the first constraint of the primal problem.

When the primal objective function and constraints are convex, Slater's conditions show that the optimal duality gap is zero. In other words, solving the dual problem and assigning point y to the base that solves $\min_i\{u_i(y) - \lambda_i^* u_i(y)/w(y)\}$ yields a primal optimal solution, for primal dual variables λ_i^*, $i = 1, \ldots, n$. Thusly motivated, in the next section we sketch the ideas behind a dual solution algorithm, and select functions $w(\cdot)$ and $q(\cdot)$ in our numerical examples that lead to a convex primal problem.

3.3 Algorithm Development

To gain some intuition over the problem with two bases, let's assume we are given an initial $\lambda_0 = (\lambda_{01}, \lambda_{02})$ for $i = 1, 2$, with resulting regions $R_1(\lambda_0)$ and $R_2(\lambda_0)$, as discussed above. As a result, one of the two things should happen:

$$\int_{R_i(\lambda_0)} q(y) u_i(y)\, dy > s_i \tag{3.1}$$

or

$$\int_{R_i(\lambda_0)} q(y) u_i(y)\, dy \leq s_i \tag{3.2}$$

Let's assume for a moment that (3.1) is correct for $i = 1$. We need to change our λ_0 on the first iteration such that, the result of the integral gets closer to s_1. For the first iteration let $\lambda_1 = (\lambda_{11}, \lambda_{12})$ where $(\lambda_{11}, \lambda_{12}) = (\lambda_{01} - \varepsilon, \lambda_{02})$ for $\varepsilon > 0$ and small. When we substitute these into the objective, we get,

$$\int_{R_1(\lambda_0)} d(y) \left[u_1(y) - \lambda_{01} \frac{u_1(y)}{w(y)} \right] dy + \int_{R_2(\lambda_0)} d(y) \left[u_2(y) - \lambda_{02} \frac{u_2(y)}{w(y)} \right] dy + s_1 \lambda_{01} + s_2 \lambda_{02} \tag{3.3}$$

and

$$\int_{R_1(\lambda_1)} d(y) \left[u_1(y) - (\lambda_{01} - \varepsilon) \frac{u_1(y)}{w(y)} \right] dy + \int_{R_2(\lambda_1)} d(y) \left[u_2(y) - \lambda_{02} \frac{u_2(y)}{w(y)} \right] dy + s_1(\lambda_{01} - \varepsilon) + s_2 \lambda_{02} \tag{3.4}$$

In order to verify we are increasing our objective, we need to verify that the difference of

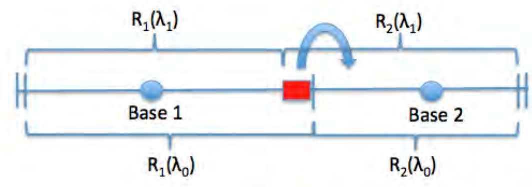

Figure 3.3: Visual representation of the allocation of the operation area

(3.4) - (3.3) is positive. With the visual help of the Figure 3.3 we can write the following as difference:

$$\int_{R_1(\lambda_1)} \varepsilon d(y)\frac{u_1(y)}{w(y)}\,dy - \int_{R_1(\lambda_0)-R_1(\lambda_1)} d(y)\left[u_1(y) - \lambda_{01}\frac{u_1(y)}{w(y)}\right] dy \\ + \int_{R_2(\lambda_1)-R_2(\lambda_0)} d(y)\left[u_2(y) - \lambda_{02}\frac{u_2(y)}{w(y)}\right] dy - \varepsilon s_1 \tag{3.5}$$

where $R_1(\lambda_0) - R_1(\lambda_1)$ and $R_2(\lambda_1) - R_2(\lambda_0)$ are defined in the sense of set differences. For simplicity, we call these set differences "red"; see Figure 3.3. Then, the last equation becomes

$$\int_{R_1(\lambda_1)} \varepsilon d(y)\frac{u_1(y)}{w(y)}\,dy - \int_{red} d(y)\left[u_1(y) - \lambda_{01}\frac{u_1(y)}{w(y)}\right] dy \\ + \int_{red} d(y)\left[u_2(y) - \lambda_{02}\frac{u_2(y)}{w(y)}\right] dy - \varepsilon s_1\,. \tag{3.6}$$

In (3.6), $\int_{red} d(y)\left[u_1(y) - \lambda_{01}\frac{u_1(y)}{w(y)}\right] dy$ and $\int_{red} d(y)\left[u_2(y) - \lambda_{02}\frac{u_2(y)}{w(y)}\right] dy$ are order of $O(\varepsilon^2)$, so we can ignore them because the leading terms are of order $O(\varepsilon)$. The remaining equation becomes

$$\int_{R_1(\lambda_1)} \varepsilon d(y)\frac{u_1(y)}{w(y)}\,dy - \varepsilon s_1\,. \tag{3.7}$$

Since $d(y) = q(y)w(y)$, Equation (3.7) can be rewritten as;

$$\int_{R_1(\lambda_1)} \varepsilon q(y)u_1(y)\,dy - \varepsilon s_1\,. \tag{3.8}$$

By assumption, (3.1) is true for $i = 1$ which, together with (3.8), implies that $\int_{R_1(\lambda_1)} q(y)u_1(y)\,dy > s_1$ for ε sufficiently small.

The case when Equation (3.2) holds for $i = 2$ is identical, and so we omit the developments. The conclusion here is that, by setting $(\lambda_{11}, \lambda_{12}) = (\lambda_{01}, \lambda_{02} + \varepsilon)$ for the first iteration, the objective of the dual problem improves as long as it holds the first constraint of the primal problem.

In summary, when Equation (3.1) holds in iteration k we set $\lambda_{k+1,i} = \lambda_{k,i} - \varepsilon$, and when Equation (3.2) holds we set $\lambda_{k+1,i} = \lambda_{k,i} + \varepsilon$ as long as the first constraint of our primal problem holds; in this way the dual objective improves and we find the optimal AO for each base.

3.4 Algorithm Formulation

The developments of the preceding section suggest the following algorithm, that we implemented in RStudio.

Data: Number of the ships on each base, location of the bases, demand on each j as quantity and importance
Result: AOs for each base
initialization;
Set every λ to zero
while *there is need to change any* λ **do**
    find $u_i(y)d_i(y) - \lambda_i u_i(y)$ for each base i
    find min of these for all i
    assign each y to base i if base i has the minimum value
    compute $\int_{\Delta_i} q(y)u_i(y)\,dy$ for each i;
    if $\int_{\Delta_i} q(y)u_i(y)\,dy > s_i\kappa$ *for any i where* κ *is allowed error* **then**
        $\lambda_i = \lambda_i - \varepsilon$;
    else
        done
    end
end

Algorithm 1: Algorithm for the model

We started with λs equal to zero because using (3.1) is easier than using (3.2) which requires checking for the first constraint of the dual problem. Starting with zero ensures that we only encounter the case for (3.1).

Starting with any value other than zero as any λ_i require us to check for Equation 3.2. Since λs are defined as $\lambda \in (-\infty, 0]$, starting with zero only requires to go toward negative infinity.

CHAPTER 4:
Analysis

In this chapter we present our model results and analyze the model effectiveness and performance.

- In Section 4.1, we analyze the basic model implication and results for different demand functions to find the AO for each base.
- In Section 4.2, we discuss the implementation of the basic model with the greedy algorithm to solve the ship allocation problem.

4.1 Basic Model

Explaining the inputs and outputs of the model would be helpful for the reader. There are three inputs of the model. They are CG demand $d(y)$ which is multiplication of $q(y)$ and $w(y)$), base locations (y_i), and number of ships on each base ($s_i(n_i)$). The first input of the basic model is CG demand, both the importance ($w(y)$) and the quantity ($q(y)$) of the demand, which are crucial for our model to work. Actually, to be able to talk about CGSB, we need to know the distribution of the demand along the area of interest. In our case, we need to capture the importance and quantity of the demand to correctly sort the missions like the CG does in real life. Also, we want our AO to be proportional to the number of ships on each corresponding base. In order to achieve that purpose, we need to capture the quantity of the demand alone and use it independently from the importance.

The second input of the model is base location (y_i). We need to know the locations of the bases to be able to solve the problem.

The third input of our model is the number of ships on each base (n_i), referred to as ship allocation. Although this will also be a result after we implement the *greedy algorithm*, to make the problem simple enough, we must assume the number of ships on each base. We calculate the AO of a given ship allocation and base locations, then using a *greedy algorithm* we check the assumptions we made about ship allocation.

Although we have the objective value as an output, the only actual output of the model is

the AO for each base. We implemented the algorithm given in Chapter 3 using RStudio, and the following parts of this chapter discuss the results of that algorithm.

The basic model, given in Chapter 3, is stated one more time as follows.

$$\begin{aligned}
\min_{\Delta_1,\ldots,\Delta_n} \sum_i \int_{\Delta_i} q(y)w(y)\sqrt{x^2+(y-y_i)^2}\,dy \\
\textbf{s.t.} \int_{\Delta_i} q(y)\sqrt{x^2+(y-y_i)^2}\,dy \le s_i \quad \forall i \\
\Delta_i \cap \Delta_k = \emptyset \quad \forall i \neq k \\
\bigcup_i \Delta_i = Y
\end{aligned} \tag{4.1}$$

where $q(y)$ is the quantity of demand for point (x,y), $w(y)$ is the importance of the demand for point (x,y) and $\sqrt{x^2+(y-y_i)^2}$ is the distance function between (x,y) and base i located at point $(0,y_i)$.

After taking the dual formulation of the problem we had much easier problem that we explained in Chapter 3 (see [4.2]).

$$\begin{aligned}
\max_{\lambda} \sum_i s_i\lambda_i + \int_0^Y f(y) \times min_i\left(u_i(y) - \lambda_i \frac{u_i(y)}{w(y)}\right) dy \\
\text{s.t. } \lambda_i \le 0 \quad \forall i
\end{aligned} \tag{4.2}$$

Using this *dual problem*, we need $u(y)-\lambda_i\,(u(y)/w(y))$ lines for each base to identify the boundary between the AOs. Boundaries will be at the point where $u(y)-\lambda_i\,(u(y)/w(y))$=$u(y)-\lambda_j\,(u(y)/w(y))$ for $i \neq j \in \{1,\ldots,n\}$.

To identify the location of the boundaries, we need to find the intersection points of the $u(y)-\lambda_i\,(u(y)/w(y))$ lines for each base where they have the equal value.

Demand Function

AOs are basically shaped by the demands for CG in the area and how these demands spread along the coastline. It is intuitive to assume that any two adjacent points have similar if not the equal demand values because they are basically in the same area. However, given

geographical differences or artificial effects made by humans, the demand functions may rise or fall suddenly. One point could be on the sea traffic line while the other is outside of it. Since no one wants to be outside of these lines, demand on the point outside the sea traffic line may be zero while the point on that line may have a high demand value.

Other examples of this sudden rise and fall can appear around exit routes from industrial ports, geographical obstacles along the coastline, or fishing area boundaries. In these cases and others, demand of the two adjacent points can have a significant difference, which causes demand functions to oscillate. As a result having a smooth demand function may not always occur, and there may be more than one intersection point on $u_i(y) - \lambda_i(u_i(y)/w(y))$ lines for each base i. This interaction point gives us the boundary between the AOs, so it is crucial for our problem.

In this case, we have several ways to choose an intersection point to determine the boundary between AOs. The first option is to use the last intersection point as the boundary, which will be biased toward the end of the plot. As shown in Figure 4.1, vertical line depicts the last intersection point. This is not ideal because it is biased towards the end point of the y-axis.

The second option is to use the average point between the first intersection point and the last as the boundary between AOs. This is also not ideal, because resulting figure might have a nonsymmetrical shape and boundary is required to be at another point that captures the asymmetric behaviour.

The last option is to use *trendlines* for each line to find the intersection point of these trendlines and thus, the AO boundaries. This way is ideal because it captures the specifics of the lines, like asymmetric behaviour that is required to be captured.

In this chapter we use these trendlines to eliminate any concerns about multiple intersection points. We refer to the intersection of these trendlines (See Figure 4.2) as the intersection point.

4.1.1 Two Bases

In this section, we discuss our sample results for two bases, as depicted in Figure 4.2. In this example, the bases are at y=30 and y=150, and there are four ships on Base 1 and six

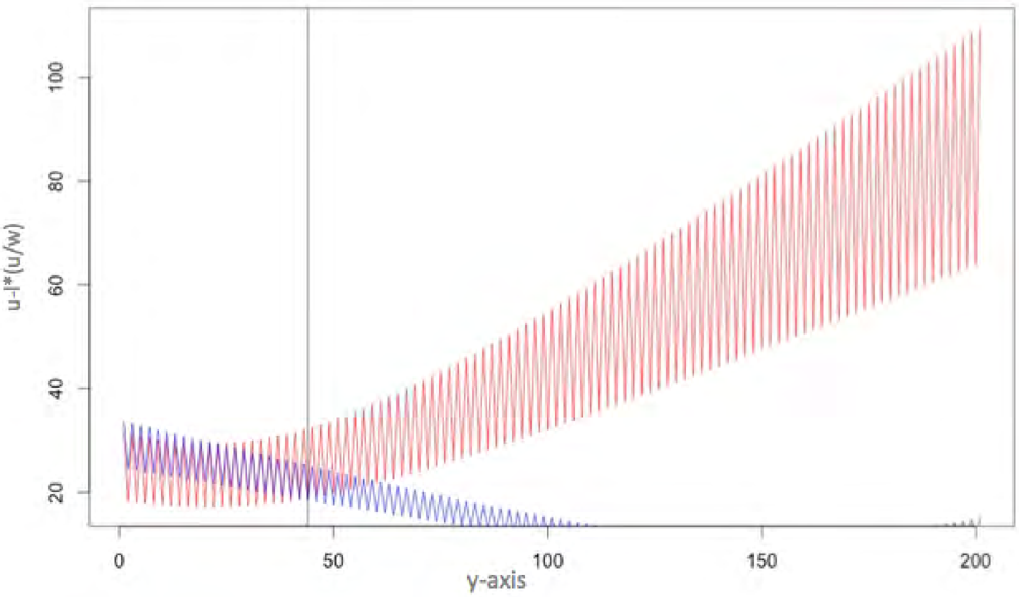

Figure 4.1: Visualization of the multiple intersection points of the $u_i(y) - \lambda_i(u_i(y)/w(y))$ lines for each base (red and blue) along the y-axis

ships on Base 2. We must state that, to have ease of computation we used constant quantity of the demand function equal to one and importance of the demand function oscillating between one and two for discretized y in every 0.1 nm. We allowed an error rate of 2%. Given this information, the basic model gives the result of an AO boundary at y=85.

Without having this tool, one might think that if importance and the quantity of the demand are constant, the optimum solution should be at y=80 with symmetric ship allocation on y-axis. The reason is the length of the coast assigned to Base 1 should be 0.4 of the interested coast line since there are four ships assigned to Base 1 out of ten total available ships.

In our case, ship allocation was not symmetric, meaning the distance between the left edge of the area and the Base 1 was length 30, but the distance between the right edge of the area and Base 2 is length 50, as shown in Figure 4.3. These distances also cause the difference which is not that clear how this will affect the result and how much it affects.

Although we do not know how much the effect will be, we can say that since Base 1 is closer to the edge of the area compared to Base 2, Base 1 can focus to use its forces more to the right side of it, which is in the middle. It results as shifting this boundary to the right

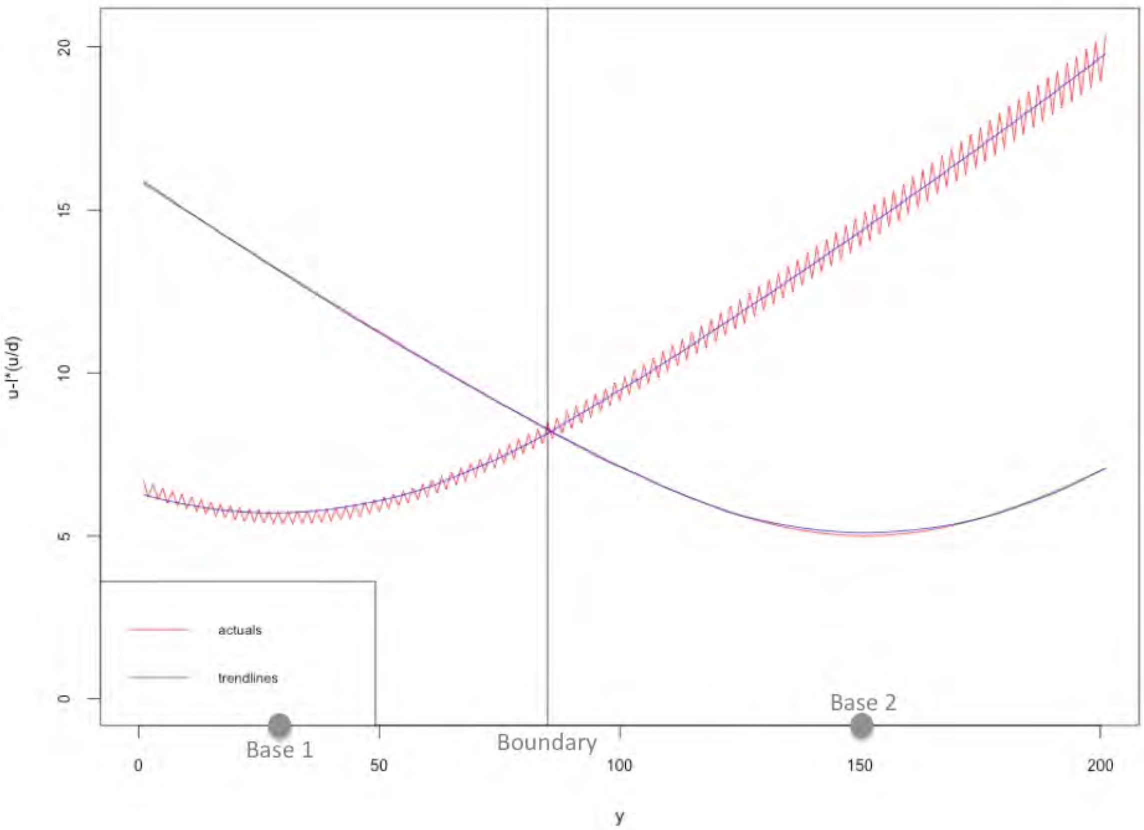

Figure 4.2: Visualization of the $u_i(y) - \lambda_i(u_i(y)/w(y))$ lines for each base which are at $y = 30$ and $y = 150$ accordingly, and 4 and 6 ships for Base 1 and 2 sequentially where the boundary is at y=85 where we used constant quantity of the demand function equals to one and importance of the demand function oscillating between one and two (2% error allowed)

but it's not easy to tell how much.

Our model shows that shift should be from y=80 to y=85. Without this tool, even with a constant demand function, it is not intuitive where to have a boundary between AOs for each base.

We showed our sample run results in Table 4.1 for two-base problems in which we use a quantity of the demand function ($q(y)$) equal to one and importance of the demand function ($d_i(y)$) oscillating between one and two. Table 4.2 also shows us the sample results for two-base problems for which we use a quantity of the demand function ($q(y)$) equal to $y/2$ and importance of the demand function ($d_i(y)$) oscillating between one and two. We allow these

Figure 4.3: Visualization of the base locations and the distances from the edges for the two-base example where bases are at y=30 and y=150 respectively

results to have a 2% error when calculating the area.

As previously mentioned, k represents the fraction of demand covered by the ships assigned to each base. In practice, we start with a small k, and increase it until right before it becomes infeasible. Since it is a fraction covered it should be less than one. However, we observe k values greater than one in Table 4.1. As long as we find a solution in which $k = 1$ there is no need to add more ships, but it depends on the values of the demand functions and we will need a different demand function which makes $k \leq 1$. We selected various demand functions, but they may appear much smaller than the real demand functions. So, we allowed the model to have any k value for different total number of ships given one demand function. That is why we end up having k values that are greater than one. Since we created these tables to see the results, we assume we always need more ships, and compare those k values as they were less than one. It will give us the ability to compare the results that we have so far.

In Table 4.1, we observe that for a total of ten ships in the first four rows, k is in range of 1.54-1.7 which is pretty close together. For nine ships in row five, k is 1.48 which is less than the previous region, and for row six, it is even less for total of seven ships. It follows the same pattern for more ships too. Thus, having more ships means more coverage over the area which is intuitive. However, objective values are not distributed systematically and it shows that ship allocation has an impact on the objective value. In Table 4.2, we observe the same pattern with one or two exceptions. Also, there are a few cases in which we have more than one boundary. These results indicate that there are three different regions, and one in the middle belongs to one base while the other at the edge belongs to the other base. In practice, it is not a desired result but having this base locations and the first constraint of the primal problem makes the model find disconnected AOs. This means we may want to change our base locations to get connected regions as the AO, but the model does find the

Table 4.1: Results for AO of bases for two-base problems in which we used constant quantity of the demand function equals to one and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
10	4-6	40-140	1750	1.7	75
10	3-7	40-140	1809	1.65	68
10	2-8	40-140	1942	1.54	48
10	5-5	40-140	1762	1.7	100
9	5-4	60-130	1815	1.48	112
7	4-3	60-130	1817	1.15	115
11	7-4	60-130	1876	1.75	129
14	9-5	60-130	1888	2.22	130
5	2-3	60-130	1805	0.83	84

Table 4.2: Results for AO of bases for two-base problems in which we used a quantity of the demand function equals to $y/20$ and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
10	4-6	40-140	9458	0.32	121
10	3-7	40-140	9094	0.33	106
10	2-8	40-140	8964	0.34	90
10	5-5	40-140	10111	0.3	134
9	5-4	60-130	10183	0.39	141-183
7	4-3	60-130	10283	0.3	142-182
11	7-4	60-130	10217	0.46	144-179
14	9-5	60-130	10217	0.59	144-179
5	2-3	60-130	9576	0.16	127

solution for the given base locations.

4.1.2 More Than Two Bases

After running our model with two bases, we want to get some intuitive results with three bases to check our model. In Figure 4.4 we can see the result of AOs for bases at $y = 10$, $y = 80$ and $y = 150$ accordingly, and 2-1-2 ships for Bases 1, 2, and 3 sequentially where the boundaries are at $y = 77$ and $y = 119$ with constant demand.

Having 2-1-2 ship allocation at Bases 1, 2, and 3 respectively, might result in having pro-

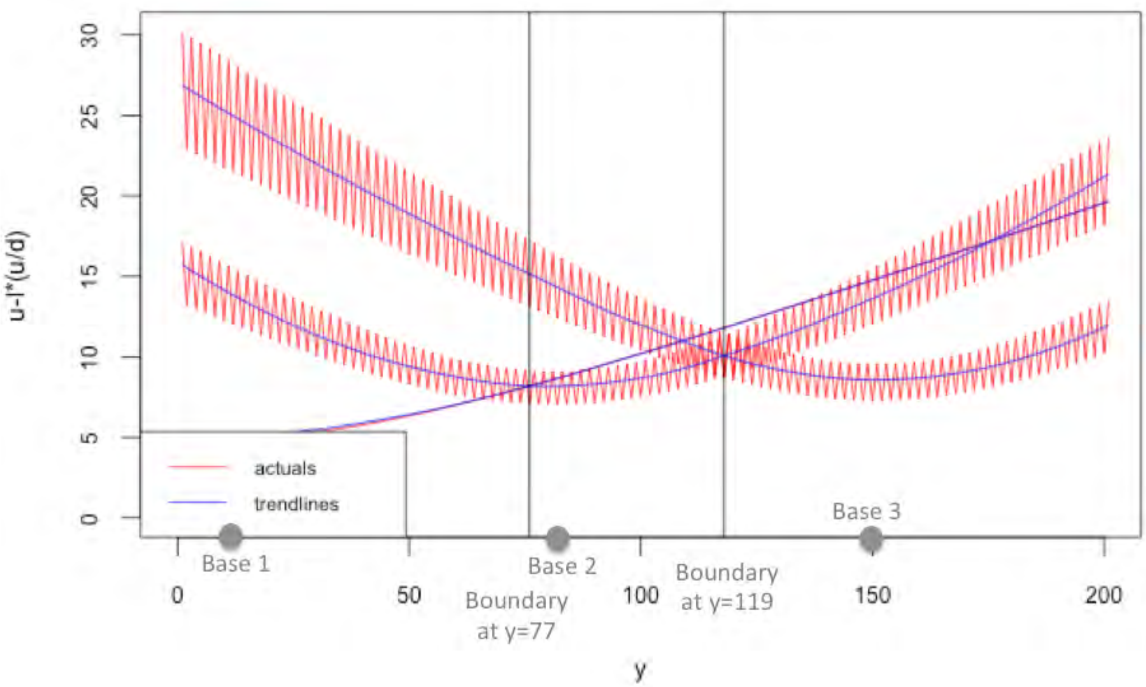

Figure 4.4: Visualization of the $u_i(y) - \lambda_i\,(u_i(y)/w(y))$ lines for each base which are at $y = 10$, $y = 80$ and $y = 150$ accordingly, and 2-1-2 ship allocations for bases one, two, and three sequentially where the boundaries are at y=77 and y=119 where we used constant quantity of the demand function equals to one and importance of the demand function oscillating between one and two (2% error allowed)

portional AOs for the bases. In this case, the lengths of the AOs should be 80, 40, and 80, which implies the boundaries should be at $y = 80$ and $y = 120$.

Even with the constant demand, as we discussed for the two-bases case, we do not expect to see the exact same boundaries. The differences between boundary lines are a length of three and one accordingly. The model matches our expectations so closely for the symmetric distribution of the bases, which gives us confidence in our model.

We showed our sample run results in Table 4.3 for three-base problems in which we used a quantity of the demand function ($q(y)$) equal to one and importance of the demand function ($w(y)$) oscillating between one and two. Table 4.4 also shows us the sample results for three base problems for which we used a quantity of the demand function ($q(y)$) equal to $|sin(y/10)|$ and importance of the demand function ($w(y)$) oscillating between one and two. We allow these results to have a 2% error when calculating the area.

In Table 4.3, we only observe for a total of ten ships, where k is in range of 1.64-1.78, which is pretty close together. Similarly, objective values are also distributed close to one another. We can say the same thing for Table 4.4. One important observation can be dramatic changes in the boundaries. A small change in the ship allocation may result in a dramatic change in the boundaries.

In Tables 4.5, 4.6 and 4.7, we give sample results for problems with four and five bases, respectively. We try to capture different demand trends with three different demand functions; constant equal to one, increasing equal to $y/20$, and oscillating equal to $|sin(y/10)|$. We chose to give intuitive samples to see if the model outputs what we expect. In Table 4.5 for the first row, since the demand function is constant and base locations are uniformly distributed, we expect to see boundaries at 50-100-150. The model give us a pretty close result since we allow it to have a 2% error. The second row gives boundaries of 41-81-121-161 which are really close to what we expected.

In Table 4.6 and 4.7 we also want to get the same results as boundaries that is why we arrange the number of ships. In Table 4.6, we have similar results, which are pretty close to what we expect. In Table 4.7, there seems to be a greater difference because we use the sinus function. We can not expect the same result with a constant demand that uses a sinus demand function.

We should state that, up until Table 4.7, we assumed ship allocation as an input rather than an output.

Table 4.3: Results for AO of bases for three-base problems where we used a quantity of the demand function equals to one and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
10	4-2-4	60-100-160	1979	1.78	79-121
10	4-4-2	60-100-160	1872	1.72	78-159
10	3-2-5	40-110-140	1878	1.74	65-104
10	5-2-3	40-110-140	2108	1.71	100-143
10	3-5-2	40-110-140	1828	1.7	65-166
10	2-6-2	40-110-140	1971	1.64	45-166

Table 4.4: Results for AO of bases for three-base problems where we used a quantity of the demand function equals to $|sin(y/10)|$ and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
10	4-2-4	60-100-160	1243	2.81	79-118
10	4-4-2	60-100-160	1185	2.75	78-155
10	2-4-4	60-100-160	1308	2.65	40-114
10	3-2-5	40-110-140	1175	2.78	68-106
10	5-2-3	40-110-140	1330	2.72	102-142
10	3-5-2	40-110-140	1146	2.73	67-168
10	2-6-2	40-110-140	1252	2.63	46-168

Table 4.5: Results for AO of bases for multiple bases problems where we used a quantity of the demand function equal to one and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
8	2-2-2-2	25-75-125-175	1583	1.51	51-101-151
10	2-2-2-2-2	20-60-100-140-180	1566	1.9	41-81-121-161

Table 4.6: Results for AO of bases for multiple bases problems where we used a quantity of the demand function equal to $y/20$ and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
16	1-3-5-7	25-75-125-175	7927	0.61	50-101-151
25	1-3-5-7-9	20-60-100-140-180	7841	0.96	40-81-121-161

Exhaustive Search

After showing the sample results, we explore the *exhaustive search* results for our model.

As we can see from Figure 4.5, we run the model for different ship assignments for each base. We used an *exhaustive search* technique, which means checking for every possible solution to find the optimal one. An interesting observation from this figure is its convexity. From the *exhaustive search* figure, it is pretty clear that having six ships on Base 1 and four ships on Base 2 has the minimum objective value that we are trying to minimize. An

Table 4.7: Results for AO of bases for multiple bases problems where we used a quantity of the demand function equal to $|sin(y/10)|$ and importance of the demand function oscillating between one and two (2% error allowed)

Total Ships	Ship Allocation	Base locations	Primal Objective Value	k	Boundary
8	2-2-2-2	25-75-125-175	1014	2.38	50-104-148
10	2-2-2-2-2	20-60-100-140-180	994	3.01	43-80-117-163

allocation of 6-4 appears to be the best allocation for constant demand and bases at $y = 50$ and $y = 170$.

4.2 Greedy Algorithm

After developing a model to define the optimal AOs for bases, we tried to answer the question of how many ships we should assign each base. We have a couple of options. These options include:

Exhaustive Search Trying every possible integer solution and finding the one that has the minimum objective to find the optimum allocation of the ships;

Greedy Algorithm Allocating one ship at a time to a base where we have the maximum decrease in the objective value; and

Non-integer Solution Finding a non-integer solution to our problem and after proving the convexity of the results, checking the nearest integer solutions to give us the optimum allocation of the ships.

Since our exhaustive search result in Figure 4.5 appears to be convex, a greedy algorithm makes more sense compared to other alternatives. One of the benefits of having a greedy algorithm is that once the results prove convexity, we are assured the globally optimal solution to our ship allocation problem.

Table 4.8 shows us the intermediate steps while solving a problem of having ten available ships for two bases located at $y = 50$ and $y = 170$. Objective values were calculated with a 2% error rate. We observe that k values constantly increase, which means we cover more of the area while objectives are close one another. As we stated before when we deal with the basic problem, objectives can also oscillate dramatically. We do not see this behavior

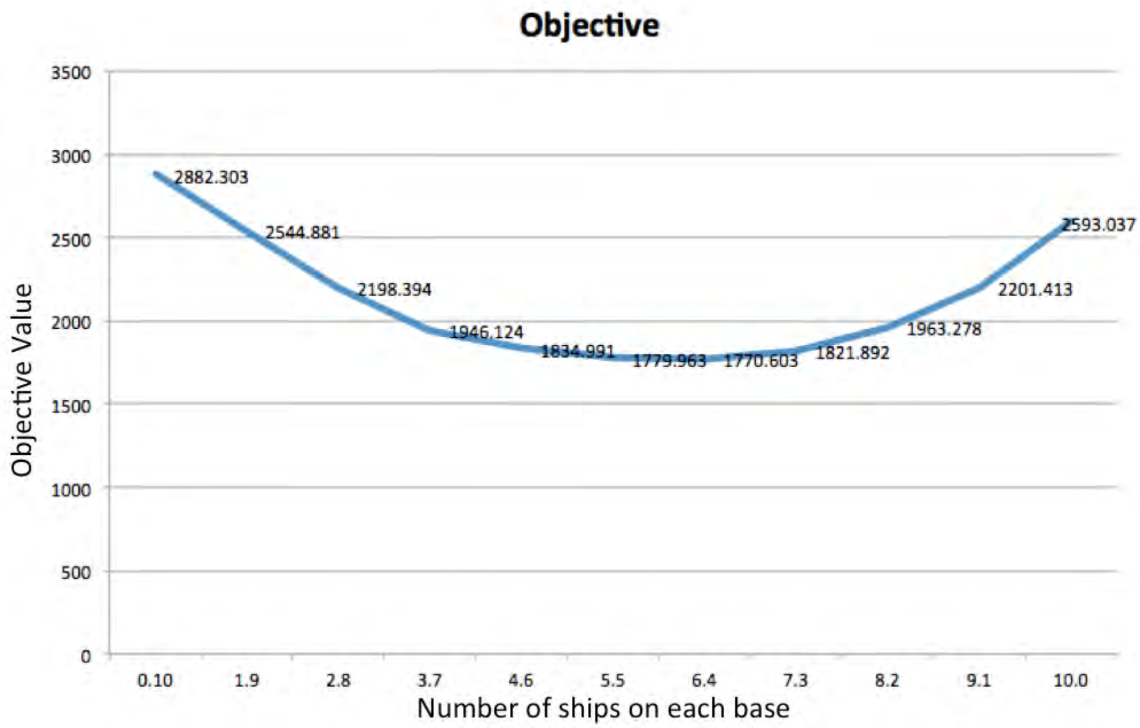

Figure 4.5: Objective Value of the different allocation of total ten ships to two different bases at $y = 50$ and $y = 170$ (Calculated with 1% error rate)

here because we are not dealing with all the objective values for different ship allocations. We only see the results where we have the minimum objective values.

In Tables 4.9 and 4.10, we give sample results for the greedy algorithm we used. The only two inputs for this table are base location and number of ships. Optimal ship allocation is one of the outputs of our model.

In Table 4.9 most results state that fair division is the best result of constant demand. However, the model gives different results if two bases are asymmetrically distributed. Though this result is intuitive, before this model, it was not clear how we would allocate ships on asymmetrically distributed bases.

In Table 4.10, we give results for the demand function equal to $y/20$. The model gives us 3-7 result for the ship allocation in most cases. Before this model, it was not clear how to divide ten ships between two bases.

Table 4.8: Depiction of the model work as we increase the available number of ships given quantity of the demand function equal to one and importance of the demand function oscillating between one and two (2% error allowed) while the two bases are located at $y = 50$ and $y = 170$

Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
3	2-1	1808	0.59	128
4	2-2	1780	0.67	104
5	3-2	1771	0.88	117
6	3-3	1780	1.01	104
7	4-3	1767	1.2	113
8	5-3	1778	1.34	121
9	5-4	1766	1.52	111
10	6-4	1771	1.68	117

As a reminder, ship allocation was a result not a given, for Tables 4.8-4.13.

In Tables 4.11, 4.12, and 4.13, we give results for four and five bases to check multiple base results with the greedy algorithm. We use the same inputs with Tables 4.5, 4.6, and 4.7, but this time, ship allocation is an output rather than an input that we assume.

The algorithm yielded the exact results in Table 4.11 as we estimated in Table 4.5. Table 4.12 uses the same inputs as 4.7, but the results are slightly different in row 1. We assume this was caused by the error we allowed in conjunction with the $|sin(y/10)|$ demand function. Table 4.13 uses the same inputs as Table 4.6, and all rows match. We also give additional results that we could not have known without this tool, which were difficult to predict beforehand.

To conclude, in this chapter we numerically tested our model under various assumptions. While we did not employ real data from a sponsor, our analysis suggests that our results are robust to different model inputs, such as importance of the demand, quantity of the demand functions, number of bases, and ships and types of ships.

Table 4.9: Sample results of the model with two bases given quantity of the demand function equal to one and importance of the demand function oscillating between one and two (2% error allowed)

Base locations	Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
50-170	10	6-4	1771	1.68	117
40-170	10	5-5	1787	1.67	103
30-170	10	5-5	1820	1.64	101
40-150	10	5-5	1745	1.76	99
50-150	10	5-5	1736	1.72	101
30-160	10	5-5	1787	1.72	99
50-180	10	6-4	1804	1.64	119
30-150	10	4-6	1771	1.68	85

Table 4.10: Sample results of the model with two bases given quantity of the demand function equal to $y/20$ and importance of the demand function oscillating between one and two (2% error allowed)

Base locations	Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
50-170	10	3-7	8745	0.34	109
40-170	10	3-7	8957	0.34	106
30-170	10	3-7	9104	0.39	103
40-150	10	2-8	8802	0.34	91
50-150	10	3-7	8705	0.4	107
30-160	10	2-8	8967	0.33	89
50-180	10	3-7	9031	0.33	110
30-150	10	2-8	8923	0.34	89

Table 4.11: Sample results of the model with more than two bases given quantity of the demand function equal to one and importance of the demand function oscillating between one and two (2% error allowed)

Base locations	Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
25-75-125-175	8	2-2-2-2	1583	1.51	51-101-151
20-60-100-140-180	10	2-2-2-2-2	1566	1.9	41-81-121-161

Table 4.12: Sample results of the model with more than two bases given quantity of the demand function equals to $|sin(y/10)|$ and importance of the demand function oscillating between one and two (2% error allowed)

Base locations	Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
25-75-125-175	8	1-3-2-2	1027	2.38	23-102-147
20-60-100-140-180	10	2-2-2-2-2	994	3.01	43-80-117-163

Table 4.13: Sample results of the model with more than two bases given quantity of the demand function equals to $y/20$ and importance of the demand function oscillating between one and two (2% error allowed)

Base locations	Available Number of Ships	Ship Allocation	Primal Objective Value	k	Boundary
25-75-125-175	16	1-3-5-7	7927	0.61	50-101-151
20-60-100-140-180	15	1-1-4-4-5	7921	0.58	51-73-127-164
20-60-100-140-180	20	1-2-5-5-7	7903	0.77	42-77-127-162
20-60-100-140-180	25	1-3-5-7-9	7841	0.96	40-81-121-161

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5: Conclusions and Future Work

In this chapter, we summarize our work, suggest future work and give some conclusions about this thesis.

5.1 Summary

We developed a model to solve a common problem that every CG faces. To the best of our knowledge, until now, there has yet to be a tool designed for finding both ship allocations and areas of operations (AOs) for each base. Therefore, developing this tool is huge step forward in this area.

We tried to capture the objective of minimizing the sum of the average demand (importance and quantity) multiplied by the distance of the point (x, y) to the assigned base. We think minimizing the distance to the assigned base, which is weighted by the importance and quantity of the demand, is appropriate. Some CGs may think that other objectives would be more appropriate like maximizing the covered demand or minimizing the response time. Thus, anyone can build a new objective and use the remaining constraints of our model to develop their own model.

Having AOs proportionate to the number of ships on each base was our most critical constraint. We controlled the AO sizes with this constraint and equalized the proportion of the coverages (k) in every AO. This also can be altered if so desired.

Other constraints in our model were *partition constraints*, which are standard in *facility location* problems, as mentioned in the literature review.

After developing the model, we used it to find AOs for each base given ship allocation, base locations, and demand functions. Then, we used a *greedy algorithm*, which basically tries to get the most utility out of the available options, to find the best ship allocation given base locations and total available number of ships.

5.2 Future Work

We intended to get data from different CG organizations to check our model, but due to time constraints we were not able to accomplish this task. Having demand data from different CG organizations and implementing them into our model could give us the understanding of how well we are doing compared to current ship allocation methods. That also helps the CG organizations to improve the utility of their resources. This task may come with challenges to the demand function. One possible problem may be unrecorded demand that we knew we could not meet so did not record. Other possible problem may be unseen activities like illegal transportations of drugs, people, etc. These challenges may require other techniques.

Our model required knowing the exact locations of the bases that were going to be used. We wanted the model to select the base locations among a certain number of candidate base locations but never had a chance to add this capability due to time limitations. Having a fixed cost for opening a base and a relation between weighted distance and this cost can give us the required results. One possible problem might occur when defining the right relation between the fixed cost for opening a base and the weighted distances. Since this would be subjective, a *sensitivity analysis* would be appropriate for this relationship.

5.3 Conclusions and Recommendations

In Chapter 4, we numerically tested our model under various assumptions. While we did not employ real data from a sponsor, our analysis suggests that our results are robust against different model inputs, such as importance of the demand, quantity of the demand functions, number of bases and ships, and types of ships.

For the case of two bases with ships already assigned, we found that k increases with the number of ships, and confirmed that our instincts aligned with our model results for the cases that we predicted the outcome. For this reason, we are confident about the model results for the ones we could not forecast the result. These conclusions carry over to problems with more than two bases. We checked up to five bases in this study.

Then, we relaxed the assumption that the ships were preassigned and used a greedy algorithm to find an efficient ship allocation. While we did not prove mathematically that the

objective function is convex in the ship allocations, extensive numerical analyses strongly indicate that convexity holds. In this case, we found that having different base locations can change the outcome, and we can also deduce how asymmetriccally located bases should have different number of ships on each base. We confirmed that our instincts aligned with our model results for the cases in which we predicted the outcome for ship allocations, too. That gives us the confidence in our model results as AO and the ship allocations for each base.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] "Mission Statement," (2014, Aug. 12). [Online]. Available: http://www.uscg.mil/hq/cg5/cg551/Mission.asp.

[2] "U.S Coast Guard Missions," (2015, Mar. 20). [Online]. Available: http://www.uscg.mil/top/missions/.

[3] "Coast Guard Station Scituate closes for season," (2014, Septmi. 08). [Online]. Available: http://coastguardnews.com/coast-guard-station-scituate-closes-for-season/2014/09/08/.

[4] "700 migrants feared dead in Mediterranean shipwreck," (2015, Apr. 19). [Online]. Available: http://www.theguardian.com/world/2015/apr/19/700-migrants-feared-dead-mediterranean-shipwreck-worst-yet.

[5] "Hundreds of migrants rescued and 10 dead in sea off Libya," (2015, May 3). [Online]. Available: http://www.usatoday.com/story/news/world/2015/05/03/italy-coast-guard-rescue-migrants-sea/26824999/.

[6] "Turk Sahil Guvenligi Guncel Faaliyetler [Turkish CG Current Events]," (2015, Apr. 28). [Online]. Available: http://www.sgk.tsk.tr/baskanliklar/genel_sekreterlik/guncel_faaliyetler.

[7] "Turk Sahil Guvenligi Guncel Faaliyetler [Turkish CG Current Events]," (2015, Apr. 28). [Online]. Available: http://www.sgk.tsk.tr/baskanliklar/genel_sekreterlik/guncel_faaliyetler/MERS%C4%B0N%2001%20MAYIS%202015.asp.

[8] "Turkish CG rescues 636 immigrants," (2015, May 5). [Online]. Available: http://www.worldbulletin.net/news/158764/turkish-coastguard-rescues-636-migrants.

[9] "Turkish CG rescues 42 immigrants," (2015, Apr. 15). [Online]. Available: http://www.sgk.tsk.tr/baskanliklar/genel_sekreterlik/guncel_faaliyetler/17%20N%C4%B0SAN%202015%20GENKUR%20ONAYLANAN%20D%C3%9CZENS%C4%B0Z%20G%C3%96%C3%87.asp.

[10] B. Olexy, private communication, May 2015.

[11] A. Federgruen and P. Zipkin, "A combined vehicle routing and inventory allocation problem," *Operations Research*, vol. 32, no. 5, pp. 1019–1037, 1984.

[12] J. G. Carlsson, "Dividing a territory among several vehicles," *INFORMS Journal on Computing*, vol. 24, no. 4, pp. 565–577, 2012.

[13] H. Min, V. Jayaraman, and R. Srivastava, "Combined location-routing problems: A synthesis and future research directions," *European Journal of Operational Research*, vol. 108, no. 1, pp. 1–15, 1998.

[14] H. W. Hamacher and Z. Drezner, *Facility location: Applications and theory*. New York, NY: Springer Science & Business Media, 2002.

[15] S. P. Fekete, J. S. Mitchell, and K. Beurer, "On the continuous Fermat-Weber problem," *Operations Research*, vol. 53, no. 1, pp. 61–76, 2005.

[16] A. Suzuki and Z. Drezner, "The minimum equitable radius location problem with continuous demand," *European Journal of Operational Research*, vol. 195, no. 1, pp. 17–30, 2009.

[17] A. Suzuki and Z. Drezner, "The p-center location problem in an area," *Location science*, vol. 4, no. 1, pp. 69–82, 1996.

[18] J. G. Carlsson and R. Devulapalli, "Dividing a territory among several facilities," *INFORMS Journal on Computing*, vol. 25, no. 4, pp. 730–742, 2012.

[19] H. Pirkul and V. Jayaraman, "A multi-commodity, multi-plant, capacitated facility location problem: Formulation and efficient heuristic solution," *Computers & Operations Research*, vol. 25, no. 10, pp. 869–878, 1998.

[20] S. Guha and S. Khuller, "Greedy strikes back: Improved facility location algorithms," *Journal of Algorithms*, vol. 31, no. 1, pp. 228–248, 1999.

[21] D. B. Shmoys, É. Tardos, and K. Aardal, "Approximation algorithms for facility location problems," in *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*. ACM, 1997, pp. 265–274.

[22] K. Aardal, F. A. Chudak, and D. B. Shmoys, "A 3-approximation algorithm for the k-level uncapacitated facility location problem," *Information Processing Letters*, vol. 72, no. 5, pp. 161–167, 1999.

[23] F. A. Chudak, *Improved Approximation Algorithms for Uncapacitated Facility Location*. New York, NY: Springer, 1998.

[24] A. Corum, "Facility location problem," (2006). [Online]. Available: http://www.tmt.unze.ba/zbornik/TMT2006/141-TMT06-433.pdf.

[25] D. Haugland, S. C. Ho, and G. Laporte, "Designing delivery districts for the vehicle routing problem with stochastic demands," *European Journal of Operational Research*, vol. 180, no. 3, pp. 997–1010, 2007.

[26] C. H. Aikens, “Facility location models for distribution planning,” *European Journal of Operational Research*, vol. 22, no. 3, pp. 263–279, 1985.

[27] J. G. Carlsson and R. Devulapalli, “Shadow prices in territory division,” *University of Minnesota*, 2013. [Online]. Available: http://www.tc.umn.edu/~{}devul002/shadowprices.pdf

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California